```
% Melissa Regalado | U29407369
% EC 401
```

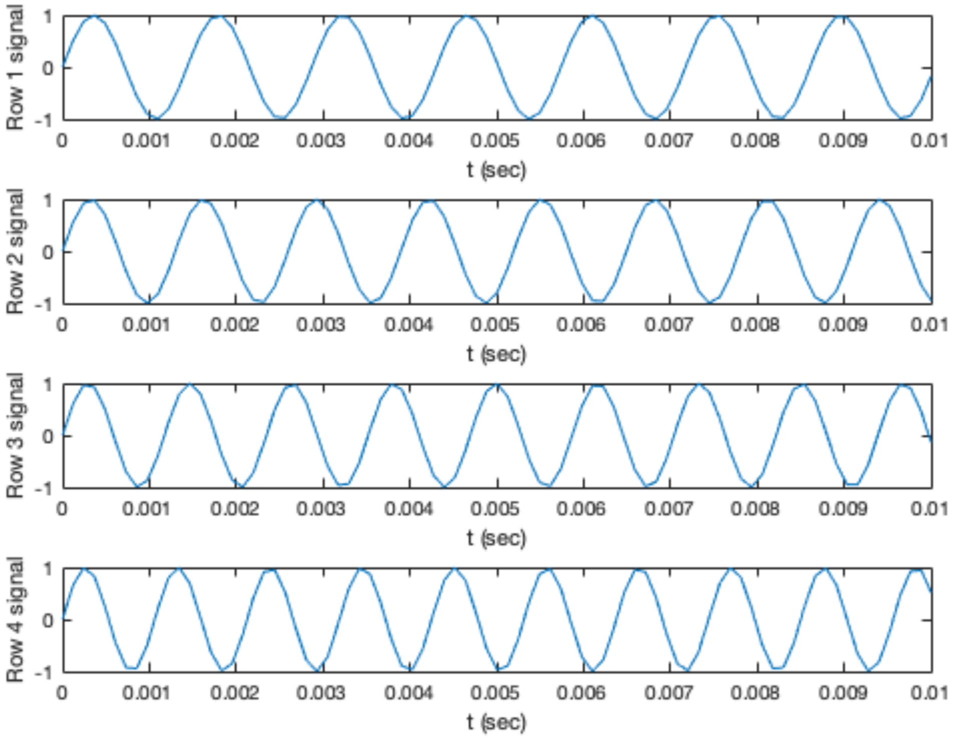
4.1

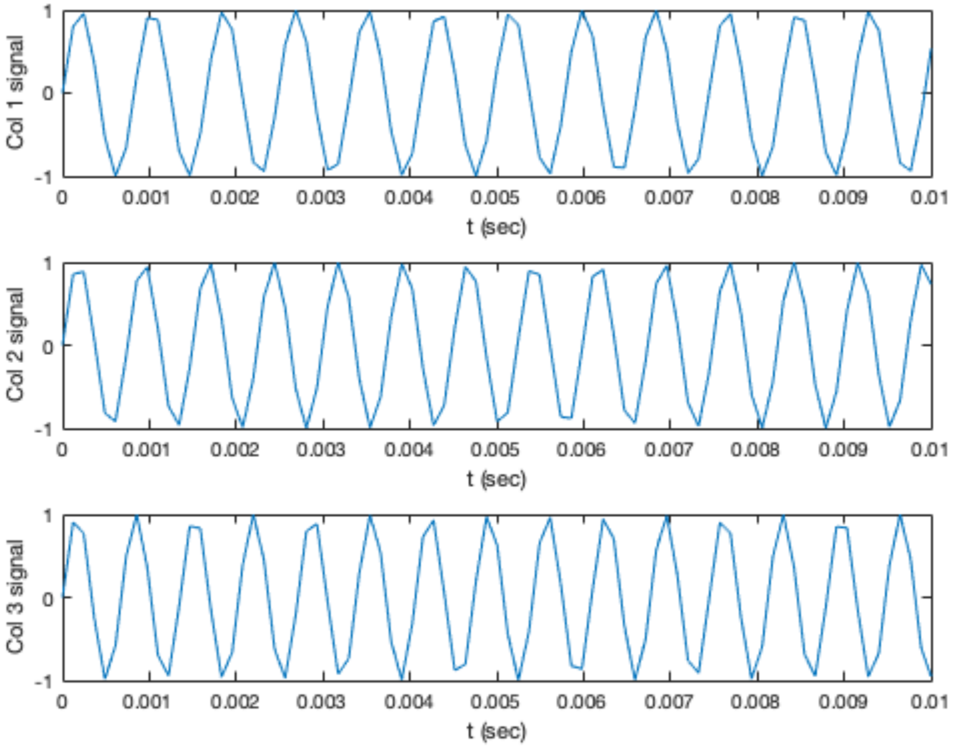
```
% Define row frequencies (Hertz)

fRow1 = 697;
fRow2 = 770 ;
fRow3 = 852;
fRow4 = 941;


% Define column frequencies (Hertz)

fCol1 = 1209;
fCol2 = 1336;
fCol3 = 1477;

% Sampling frequency and duration
% 8.192 kHz
Fs = 8192;
T = 1;
N = round(Fs*T);
t = (0:N-1)/Fs;

% create row signals

sRow1 = sin(2*pi*fRow1*t);
sRow2 = sin(2*pi*fRow2*t);
sRow3 = sin(2*pi*fRow3*t);
sRow4 = sin(2*pi*fRow4*t);

% column

sCol1 = sin(2*pi*fCol1*t);
sCol2 = sin(2*pi*fCol2*t);
sCol3 = sin(2*pi*fCol3*t);


%Plotting row signals

figure(1)
subplot(4,1,1);
plot(t,sRow1);
xlabel("t (sec)");
ylabel("Row 1 signal");
axis([0 0.01 -1 1]);


figure(1)
subplot(4,1,2);
```

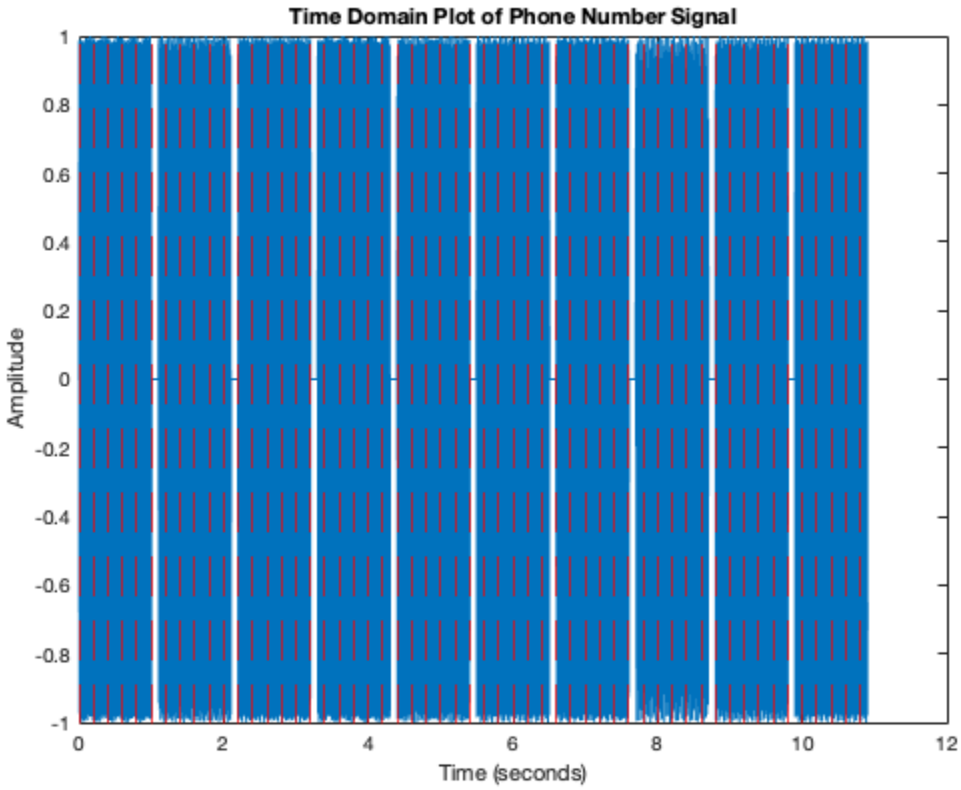
```
plot(t,sRow2);
xlabel("t (sec)");
ylabel("Row 2 signal");
axis([0 0.01 -1 1]);

figure(1)
subplot(4,1,3);
plot(t,sRow3);
xlabel("t (sec)");
ylabel("Row 3 signal");
axis([0 0.01 -1 1]);

figure(1)
subplot(4,1,4);
plot(t,sRow4);
xlabel("t (sec)");
ylabel("Row 4 signal");
axis([0 0.01 -1 1]);

%plot columns

figure(2)
subplot(3,1,1);
plot(t,sCol1);
xlabel("t (sec)");
ylabel("Col 1 signal");
axis([0 0.01 -1 1]);

figure(2)
subplot(3,1,2);
plot(t,sCol2);
xlabel("t (sec)");
ylabel("Col 2 signal");
axis([0 0.01 -1 1]);

figure(2)
subplot(3,1,3);
plot(t,sCol3);
xlabel("t (sec)");
ylabel("Col 3 signal");
axis([0 0.01 -1 1]);


% play row signals
sound(sRow1,Fs);
pause(0.1);

sound(sRow2,Fs);
pause(0.1);

sound(sRow3,Fs);
pause(0.1);

sound(sRow4,Fs);
```

```
pause(0.1);

% play column
sound(sCol1,Fs);
pause(0.1);

sound(sCol2,Fs);
pause(0.1);

sound(sCol3,Fs);
pause(0.1);


% Dual tone signals
% each represent a keypad
s1 = (sRow1+sCol1)/2;
s2 = (sRow1+sCol2)/2;
s3 = (sRow1+sCol3)/2;

s4 = (sRow2+sCol1)/2;
s5 = (sRow2+sCol2)/2;
s6 = (sRow2+sCol3)/2;

s7 = (sRow3+sCol1)/2;
s8 = (sRow3+sCol2)/2;
s9 = (sRow3+sCol3)/2;

sStar = (sRow4 + sCol1)/2;
s0=(sRow4+sCol2)/2;
sPound = (sRow4 + sCol3)/2;

%playing each key with 0.2s pause

sound(s1,Fs);
pause(0.2);
sound(s2,Fs);
pause(0.2);
sound(s3,Fs);
pause(0.2);

sound(s4,Fs);
pause(0.2);
sound(s5,Fs);
pause(0.2);
sound(s6,Fs);
pause(0.2);

sound(s7,Fs);
pause(0.2);
sound(s8,Fs);
pause(0.2);
sound(s9,Fs);
pause(0.2);
```

```
sound(sStar,Fs);
pause(0.2);
sound(s0,Fs);
pause(0.2);
sound(sPound,Fs);
pause(0.2);


%b.3)

Tpause = 0.1 ;
p = zeros(1,round(Fs * Tpause));

% phone number
sNumber = [s7 p s0 p s8 p s7 p s3 p s1 p s9 p s6 p s5 p s1];

%play phone number
sound(sNumber,Fs);

% plot and time vector
tNumber = [0:(length(sNumber)-1)] / Fs;

figure(3)
plot(tNumber,sNumber);
xlabel("Time (seconds)");
ylabel("Amplitude");
title('Time Domain Plot of Phone Number Signal');

hold on ;

digit = 0:0.2:(length(sNumber)-1)/Fs;

for i = 1:length(digit)
    xline(digit(i),"--r");
end
hold off;
```


Row 1 signal
Row 2 signal
Row 3 signal
Row 4 signal
1
0
-1
0
0.001
0.002
0.003
0.004
0.005
0.006
0.007
0.008
0.009
0.01
t (sec)


Col 1 signal
Col 2 signal
Col 3 signal
1
0
-1
0
0.001
0.002
0.003
0.004
0.005
0.006
0.007
0.008
0.009
0.01
t (sec)



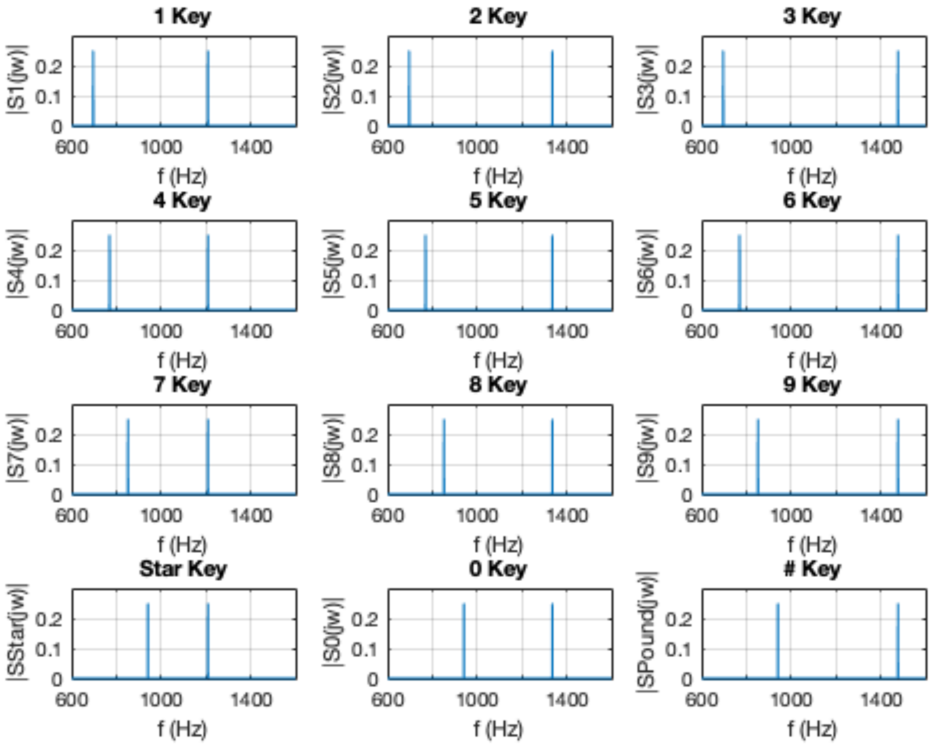
4.2

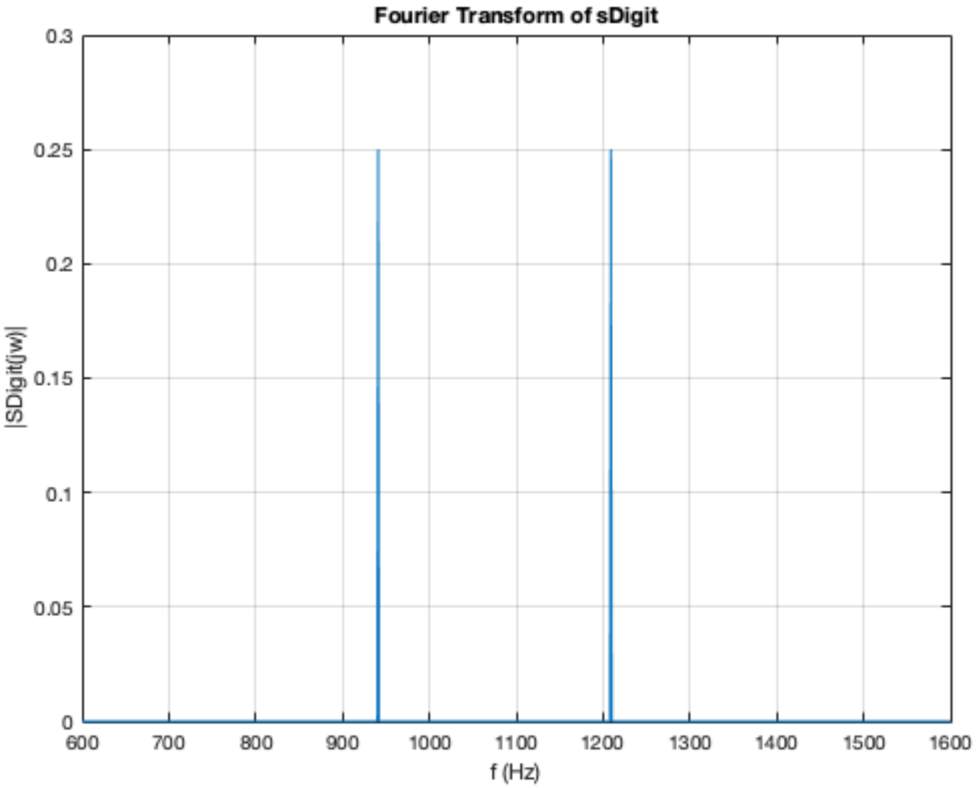
```
cd('/Users/melissaregalado/Documents/MATLAB/EC401/');


% FT DTMF signal
[S1, f] = ctft(s1, Fs);
[S2, f] = ctft(s2, Fs);
[S3, f] = ctft(s3, Fs);

[S4, f] = ctft(s4, Fs);
[S5, f] = ctft(s5, Fs);
[S6, f] = ctft(s6, Fs);

[S7, f] = ctft(s7, Fs);
[S8, f] = ctft(s8, Fs);
[S9, f] = ctft(s9, Fs);

[S0, f] = ctft(s0, Fs);
[SStar, f] = ctft(sStar, Fs);
[SPound, f] = ctft(sPound, Fs);

figure(4);

subplot(4,3,1);
```

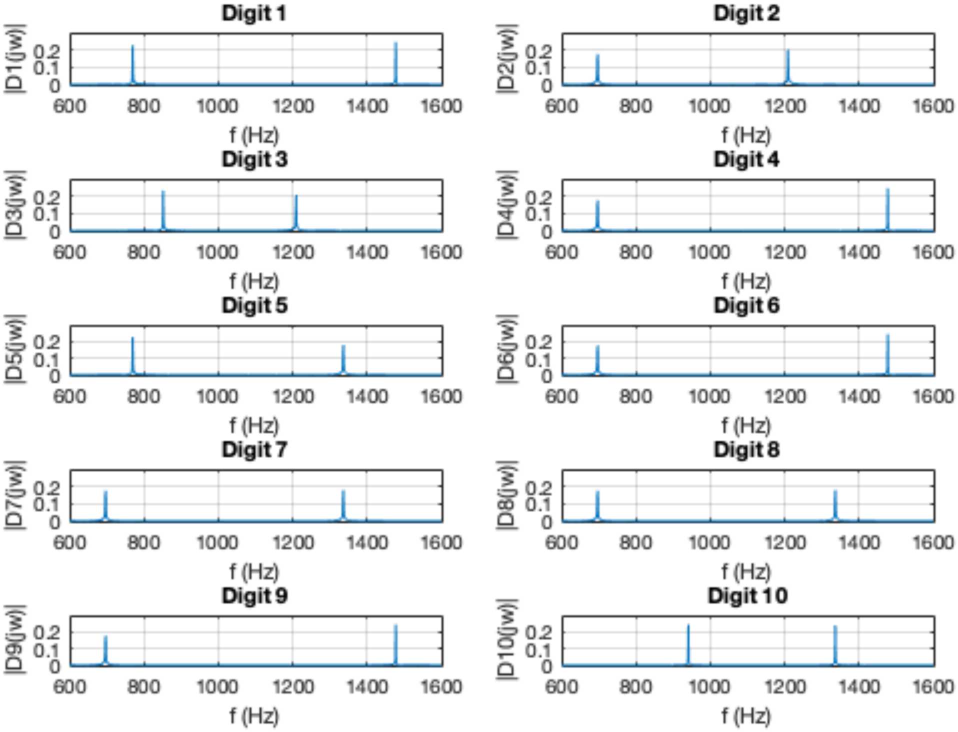
```
plot(f,abs(S1));
grid
xlabel('f (Hz)');
ylabel('|S1(jw)|');
title('1 Key');
axis([600 1600 0 0.3]);
subplot(4,3,2);
plot(f,abs(S2));
grid
xlabel('f (Hz)')
ylabel('|S2(jw)|')
title('2 Key')
axis([600 1600 0 0.3]) ;
subplot(4,3,3);
plot(f,abs(S3));
grid
xlabel('f (Hz)')
ylabel('|S3(jw)|')
title('3 Key')
axis([600 1600 0 0.3]) ;

subplot(4,3,4);
plot(f,abs(S4));
grid
xlabel('f (Hz)')
ylabel('|S4(jw)|')
title('4 Key')
axis([600 1600 0 0.3])
subplot(4,3,5);
plot(f,abs(S5));
grid
xlabel('f (Hz)')
ylabel('|S5(jw)|')
title('5 Key')
axis([600 1600 0 0.3])
subplot(4,3,6);
plot(f,abs(S6));
grid
xlabel('f (Hz)')
ylabel('|S6(jw)|')
title('6 Key')
axis([600 1600 0 0.3])

subplot(4,3,7);
plot(f,abs(S7));
grid
xlabel('f (Hz)')
ylabel('|S7(jw)|')
title('7 Key')
axis([600 1600 0 0.3])
subplot(4,3,8);
plot(f,abs(S8));
grid
xlabel('f (Hz)')
```

```
ylabel('|S8(jw)|')
title('8 Key')
axis([600 1600 0 0.3])
subplot(4,3,9);
plot(f,abs(S9));
grid
xlabel('f (Hz)')
ylabel('|S9(jw)|')
title('9 Key')
axis([600 1600 0 0.3])

subplot(4,3,10);
plot(f,abs(SStar));
grid;
xlabel('f (Hz)')
ylabel('|SStar(jw)|')
title('Star Key')
axis([600 1600 0 0.3])
subplot(4,3,11);
plot(f,abs(S0));
grid;
xlabel('f (Hz)')
ylabel('|S0(jw)|')
title('0 Key')
axis([600 1600 0 0.3])
subplot(4,3,12);
plot(f,abs(SPound));
grid;
xlabel('f (Hz)')
ylabel('|SPound(jw)|')
title('# Key')
axis([600 1600 0 0.3])

%b
load decode(1).mat;

[SDigit,f]=ctft(sDigit,Fs);


figure(5);
plot(f,abs(SDigit));
grid;
xlabel('f (Hz)');
ylabel('|SDigit(jw)|');
title('Fourier Transform of sDigit');
axis([600 1600 0 0.3]);


[nstart, nstop] = dtmfcut(sNumber, Fs);

d1 = sNumber(nstart(1):nstop(1));
d2 = sNumber(nstart(2):nstop(2));
d3 = sNumber(nstart(3):nstop(3));
d4 = sNumber(nstart(4):nstop(4));
```

```
d5 = sNumber(nstart(5):nstop(5));
d6 = sNumber(nstart(6):nstop(6));
d7 = sNumber(nstart(7):nstop(7));
d8 = sNumber(nstart(8):nstop(8));
d9 = sNumber(nstart(9):nstop(9));
d10 = sNumber(nstart(10):nstop(10));

digits = {d1, d2, d3, d4, d5, d6, d7, d8, d9, d10};
digit_labels = {'Digit 1', 'Digit 2', 'Digit 3', 'Digit 4', 'Digit 5','Digit
6', 'Digit 7', 'Digit 8', 'Digit 9', 'Digit 10'};

figure(6)

for i = 1:10
    % same number of samples
    Nsamp = length(digits{i});
    [D, f] = ctft(digits{i}, Fs, Nsamp);

    % Align lengths
    min_len = min(length(f), length(D));
    f = f(1:min_len);
    D = D(1:min_len);

    % plot in a subplot
    subplot(5, 2, i);
    plot(f, abs(D)); grid;
    xlabel('f (Hz)');
    ylabel(['|D', num2str(i), '(jw)|']);
    title(digit_labels{i});
    axis([600 1600 0 0.3]);
end
```


1 Key
|S1(jw)|
2 Key
|S2(jw)|
3 Key
|S3(jw)|
4 Key
|S4(jw)|
5 Key
|S5(jw)|
6 Key
|S6(jw)|
7 Key
|S7(jw)|
8 Key
|S8(jw)|
9 Key
|S9(jw)|
Star Key
|SStar(jw)|
0 Key
|S0(jw)|
# Key
|SPound(jw)|
0
0.1
0.2
600
1000
1400
f (Hz)

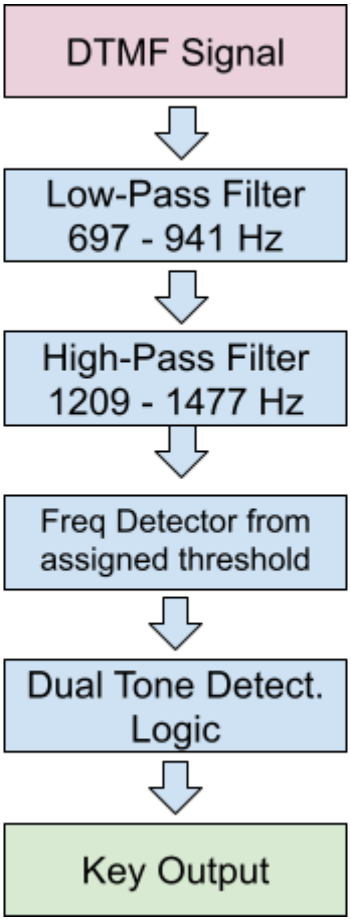

Fourier Transform of sDigit
|SDigit(jw)|
0
0.05
0.1
0.15
0.2
0.25
0.3
600
700
800
900
1000
1100
1200
1300
1400
1500
1600
f (Hz)



Digit 1 Digit 2 Digit 3 Digit 4 Digit 5 Digit 6 Digit 7 Digit 8 Digit 9 Digit 10

Published with MATLAB® R2024b

An automatic DTMF decoder can be designed by using LTI systems with tuned bandpass filters to set a particular frequency. The input signals will pass through these filters and detect frequency pairs to their corresponding keys. In regards to noise, low-pass filters can be implemented to smooth rapid fluctuations of signal. Low-pass filters can also be used to better identify a threshold frequency to determine if the system will perform or not.


DTMF Signal
Low-Pass Filter 697 - 941 Hz
High-Pass Filter 1209 - 1477 Hz
Freq Detector from assigned threshold
Dual Tone Detect. Logic
Key Output